

UNIVERSIDADE COMUNITÁRIA REGIONAL DE CHAPECÓ

Jhony Maiki Maseto

UMA ANÁLISE ENTRE FRAMEWORKS DE PHP

Chapecó-SC, 2006

JHONY MAIKI MASETO

UMA ANÁLISE ENTRE FRAMEWORKS DE PHP

Monografia de trabalho de conclusão de curso  
apresentada à Unochapecó como parte dos requisitos para  
obtenção do grau de bacharel em Ciência da computação  
em 2006.

Orientador: Elton Luís Minetto.

Chapecó-SC junho.2006

## Índice de Tabelas

<i>Tabela 1: (the PHP Group.2006)</i>	13
<i>Tabela 2: (Dall'Oglio, Pablo.2005).</i>	17
<i>Tabela 3: (Dall'Oglio, Pablo.2005).</i>	18
<i>Tabela 4: (Dall'Oglio, Pablo.2005).</i>	19
<i>Tabela 5: (Morais, João Cruz.2004).</i>	20
<i>Tabela 6: (Morais, João Cruz.2004).</i>	21
<i>Tabela 7: (Morais, João Cruz.2004).</i>	21
<i>Tabela 8: (Lozano,Fernando.2002).</i>	25
<i>Tabela 9: (Lozano,Fernando.2002).</i>	25
<i>Tabela 10 : (Comunidade Cyaneus.2006),(The Pallett Group.2005).</i>	34
<i>Tabela 11( Procedimentos Metodológicos).</i>	44

## Sumário

<b>1 .TEMA</b>	<b>6</b>
<b>2 .DELIMITAÇÃO DO PROBLEMA</b>	<b>7</b>
<b>3 .QUESTÕES DE PESQUISA</b>	<b>8</b>
<b>4 .JUSTIFICATIVA</b>	<b>9</b>
<b>5 .OBJETIVOS</b>	<b>10</b>
<b>4.1 Geral:</b>	<b>10</b>
<b>4.2 Específicos:</b>	<b>10</b>
<b>6 .FUNDAMENTAÇÕES TEÓRICAS</b>	<b>11</b>
<b>6.1 FRAMEWORKS</b>	<b>11</b>
6.1.1 INTRODUÇÃO	11
<b>6.2 PHP</b>	<b>12</b>
6.2.1 Introdução	12
6.2.1.1 Vantagens em Utilizar o PHP:	14
6.2.1.2 Desvantagens no uso do PHP:	15
6.2.2 PHP4	16
6.2.3 PHP5	17
6.2.3.1 Orientação Objeto e conceitos	17
<b>6.3 QUALIDADE DE SOFTWARE.</b>	<b>21</b>
6.3.1 Qualidade do processo.	22
6.3.2 Qualidade do produto.	23

6.3.2.1 Testes de verificação	23
6.3.2.2 Testes de validação	24
6.3.3 PHPUnit - unit testing framework for PHP	24
<b>6.4 FRAMEWORKS PHP</b>	<b>26</b>
6.4.1 PRADO - PHP 5 Framework	27
6.4.2 Symfony Framework	29
6.4.3 Cake PHP Framework	31
6.4.4 Requisitos de uso dos frameworks	33
<b>6.5 TÓPICOS PARA COMPARAÇÃO</b>	<b>33</b>
6.5.1 Tabela comparativa	34
6.5.1.1 Mvc	34
6.5.1.2 Multiples DB's	35
6.5.1.3 ORM	35
6.5.1.4 Validation	35
6.5.1.5 AJAX	36
6.5.1.6 Auth Module	36
6.5.1.7 Modules	36
6.5.1.8 Tableless	37
6.5.1.9 DB Objects	37
6.5.1.10 Templates	37
6.5.1.11 Caching	37
6.5.1.12 Internacionalização	38
6.5.1.13 Integração Pear	39
5.1.14 Documentação detalhada	39
<b>7 .CRONORAMA DE AÇÃO</b>	<b>40</b>
<b>8 .ORÇAMENTO</b>	<b>42</b>
<b>9 .PROCEDIMENTOS METODOLÓGICOS</b>	<b>43</b>
<b>10 .REFERÊNCIAS BIBLIOGRÁFICAS</b>	<b>45</b>

# **1 .TEMA**

Uma análise entre Frameworks de PHP5.

## **2 .DELIMITAÇÃO DO PROBLEMA**

Tendo em vista a grande quantidade de frameworks disponíveis hoje, e o pouco tempo tempo que o desenvolvedor dispõe para analisar qual dos frameworks se encaixa melhor ao seu tipo de aplicação, bem como a falta de documentação existente em muitos casos estamos propondo, fazer uma análise comparativa entre, no mínimo, dois frameworks de php com o objetivo de verificar qual apresenta-se melhor em relação aos quesitos a serem analisados ao decorrer do projeto.

Para facilitar a definição de qual é o framework mais indicado será desenvolvido um protótipo usando-se os frameworks para melhorar a analise na pratica.

### **3 .QUESTÕES DE PESQUISA**

- 1) O que são Frameworks de PHP, quais são as suas utilidades e como funcionam ?
- 2) O que é PHP, de onde surgiu e o que PHP contempla ?
- 3) O que é programação orientada a objetos, e como suas técnicas se aplicam em php?
- 4) Como testar a qualidade de um software?
- 5) Prado Framework, o que é e como funciona?
- 6) Symfony Framework, o que é e como ele funciona?
- 7) Cake Framework , o que é e como ele funciona?
- 8) Qual Framework é a melhor opção para o desenvolvimento de novos aplicativos e sites?

## **4 .JUSTIFICATIVA**

Colaborar com a comunidade desenvolvedora no sentido de descrever qual dos softwares se encontra em um estado mais maduro e seguro para se desenvolver aplicações, utilizando a estrutura dos frameworks de PHP. Também colaborar com os desenvolvedores dos próprios frameworks mostrando -lhes o que os seus frameworks não estão contemplando e que seria de interesse da comunidade que fosse embutido em seu projeto, tendo como propósito incentivar o uso da metodologia de desenvolvimento com frameworks entre a comunidade acadêmica.

Esta pesquisa tem o intuito de indicar aprimoramentos das ferramentas de trabalho dos programadores PHP. Descreve-se neste um comparativo entre alguns frameworks, com o objetivo de auxiliar na escolha da melhor ferramenta de framework para cada tipo de aplicação.

## **5 .OBJETIVOS**

### **4.1 Geral:**

Fazer uma análise comparando dois ou mais frameworks de PHP, bem como definir qual é o mais indicado para desenvolvimento de aplicações, explanando amplamente todos os conceitos abordados pelos frameworks.

### **4.2 Específicos:**

Estudar dois ou mais frameworks de php e fazer uma avaliação segundo alguns quesitos para ver qual deles é o mais indicado para o uso, bem como descrever o que cada um deles contempla.

- Definir o que é php como ele surgiu, seu funcionamento, e onde ele é usado.
- Estudar os conceitos de qualidade de software para definir os quesitos a serem analisados.
- Identificar vantagens e desvantagens do uso de frameworks.
- Identificar o framework mais eficiente para desenvolvimento de aplicativos php.

## **6 .FUNDAMENTAÇÕES TEÓRICAS**

### **6.1 FRAMEWORKS**

#### **6.1.1 INTRODUÇÃO**

Com a necessidade de reutilização de código pelos programadores obteve-se o conceito de criação de sistemas a partir de códigos e objetos já escritos. Com a reunião deste grupo de códigos e objetos obteve-se então o conceito de framework. Desta forma, o princípio de todo framework é ser uma solução re-usável, estável e bem documentada.

De acordo com (FAYAD, W.E.; SCHMIDT, D.C.; JOHNSON, R.E,1999 apud De Souza,Marcos Vinícius Bittencourt ,2004), “Um framework consiste em um conjunto de classes que se relacionam e representam uma solução incompleta. Um framework é o esqueleto de uma aplicação que pode ser customizado por um desenvolvedor da aplicação”

Ainda de acordo com (JOHNSON, R.; FOOTE,1998 apud De Souza,Marcos Vinícius Bittencourt ,2004) , “Um framework orientado a objetos é um projeto re-utilizável de software definido por um conjunto de classes abstratas e pela maneira pela qual as instâncias dessas classes colaboram entre si”.

No desenvolvimento do software um framework, pode ser considerado uma estrutura de suporte definida em que um outro projeto do software pode ser organizado e desenvolvido. Tipicamente, um framework pode incluir programas de apoio, bibliotecas de código, linguagens de script e outros softwares para ajudar a desenvolver e juntar diferentes componentes do seu projeto. (WIKIPÉDIA.2006).

Especificamente em orientação a objeto, framework é um conjunto de classes com objetivo de reutilização de um design, provendo um guia para uma solução de arquitetura em um domínio específico de software (WIKIPÉDIA.2006).

Framework se diferencia de uma simples biblioteca (*toolkit*)<sup>1</sup>, pois esta se concentra apenas em oferecer implementação de funcionalidades, sem definir a reutilização de uma solução de arquitetura (*design*). (WIKIPÉDIA.2006).

## 6.2 PHP

### 6.2.1 Introdução

PHP (um acrônimo<sup>2</sup> recursivo para "PHP: Hypertext Preprocessor") é uma linguagem de script Open Source de uso geral, muito utilizada e especialmente guarnecida para o desenvolvimento de aplicações Web embutível dentro do HTML. (PHP.NET.2005).

O PHP é uma linguagem de programação de computadores interpretada, livre e muito utilizada para gerar conteúdo dinâmico na Web. Apesar de ser uma linguagem de fácil aprendizado e de uso para pequenos scripts dinâmicos e simples, o PHP é uma linguagem poderosa orientada à objetos. (WIKIPÉDIA.2006).

A linguagem surgiu por volta de 1994, como um subconjunto de scripts Perl criados por Ramus Lerdof. Com as adições de Zeev Suraski e

---

1 Toolkit é um conjunto de elementos básicos para construção de software. Normalmente são implementados como uma biblioteca de rotinas ou uma plataforma para aplicativos.

2 Um acrônimo ou sigla é um agrupamento das iniciais de várias palavras. Acrônimos recursivos são acrônimos onde a expansão inclui o próprio termo, como na definição de funções recursivas. PHP: Hypertext Pre-Processor (Originalmente, PHP significava Personal HomePage)

Andi Gutmans, dois programadores israelitas pertencentes ao Technion<sup>1</sup> o instituto israelita de tecnologia, que reescreveram o parser<sup>2</sup>, era lançada em 1997 a PHP 3, primeira versão estável e parecida com a linguagem atual. Ao reescrever o parser, foi criado o Zend Engine, que é mantido oficialmente pela empresa Zend em conjunto com a comunidade PHP. Em maio de 2000 veio a público a versão 4, e em julho de 2004, a versão 5, onde a principal mudança foi uma nova API<sup>3</sup> para orientação a objetos provida pelo Zend Engine2.(WIKIPÉDIA.2006).

Trata-se de uma linguagem extremamente modularizada, o que a torna ideal para instalação e uso em servidores web. Diversos módulos são criados no repositório de extensões PECL (PHP Extension Community Library) e alguns destes módulos são introduzidos como padrão em novas versões da linguagem. É muito parecida, em tipos de dados, sintaxe e mesmo funções, com a linguagem C e com a C++. Pode ser, dependendo da configuração do servidor, embutida no código HTML. Além disso, destaca-se a extrema facilidade com que PHP lida com servidores de base de dados, como MYSQL, PostgreSQL, Microsoft SQL Server e Oracle. (WIKIPÉDIA. 2006).

Quanto ao modo de execução dos scripts PHP, destaca-se:

O que distingue o PHP de algo como Javascript no lado do cliente é que o código é executado no servidor. Se você tivesse um script similar ao do exemplo descrito abaixo em seu servidor, o cliente receberia os resultados da execução desse script, sem nenhum modo de determinar como é o código fonte. Você pode inclusive configurar seu servidor para processar todos os seus arquivos HTML como PHP, e então não haverá nenhum modo dos usuários descobrirem que se você usa essa linguagem ou não. (The PHP Group.2006).

Um exemplo introdutório:

```
<html>
  <head>
    <title>Exemplo</title>
  </head>
  <body>
    <?php
      echo "Olá, Eu sou um script PHP!"; //O servidor executa este comando
    ?>                                     // no momento do carregamento da página.
  </body>
</html>
```

Tabela 1: (the PHP Group.2006)

<sup>1</sup> O Technion - Instituto Israelita de Tecnologia - é uma universidade em Haifa, Israel. Fundada em 1924, é a universidade mais antiga de Israel. O Technion é famoso pela ciência e engenharia, mas também oferece bons cursos de arquitetura e medicina, entre outros.

<sup>2</sup> Um Parser é um programa de computador (ou apenas um componente de um programa) que serve para analisar a estrutura gramatical de uma entrada

<sup>3</sup> API, de Application Programming Interface (ou Interface de Programação de Aplicativos) é um conjunto de rotinas e padrões estabelecidos por um software para utilização de suas funcionalidades por programas aplicativos -- isto é: programas que não querem envolver-se em detalhes da implementação do software, mas apenas usar seus serviços.

Existem iniciativas para utilizar o PHP como linguagem de programação de sistemas cliente-servidor. A mais notável é a PHP-GTK. Define-se como um conjunto do PHP com a biblioteca, portada do C++ , GTK fazendo assim softwares inter-operacionais entre Windows e Linux.(WIKIPÉDIA.2006).

#### 6.2.1.1 Vantagens em Utilizar o PHP:

Segundo SOUSA.,Sandro(2005) as principais vantagens do PHP são:

- Licença gratuita
- Plataforma (SO) gratuita para se rodar ele (GNU/Linux)
- Velocidade de processamento
- Eficiência de processamento
- Métodos de segurança eficientes
- Plataforma
- Roda em qualquer tipo de plataforma (SO)
- Código fonte livre
- Exceptions (para controle de fluxo)
- Orientação a objetos
- É a linguagem Web mais popular e que mais cresce (em ritmo bem acelerado) no mercado segundo netcraft.
- Possibilita a utilização dos maiores e mais utilizados Bancos de dados no mercado (Adabas D, InterBase, PostgreSQL, dBase, FrontBase, SQLite, Empress, mSQL, Solid, FilePro, Direct MS-SQL Sybase, Hyperwave, MySQL, Velocis, IBM DB2,

ODBC, Unix dbm, Informix, Oracle (OCI7 e OCI8), Ingres, Ovrimos, Firebird) sem necessitar de configuração externa como no caso do ASP que necessita configurar o ODBC

- Esta sempre em atualização e tendo corrigidas falhas e adicionados novos recursos
- É mais estável e exige e consome menos recursos de hardware do servidor
- Flexibilidade
- Componentes nativos, não dependendo de componentes externos para algumas funcionalidades básicas (como é o caso do ASP aonde muitos desses componentes são até pagos)
- Documentação, controle e reportamento de erros
- Comunidade de desenvolvimento super participativa e prestativa
- Planos de hospedagem Web (na grande maioria dos casos) mais baratos e sem nenhum custo extra para a utilização do MySQL em conjunto com o PHP
- O Apache (servidor Web utilizado para rodar o PHP) é bem seguro, já o IIS (utilizado para rodar o ASP) tem diversas brechas de segurança.

#### 6.2.1.2 Desvantagens no uso do PHP:

Mesmo com uma grande lista de vantagens de PHP como a acima citada alguns autores consideram como desvantagens:

- Segundo (Canal Html.2006), há uma centralização incômoda das variáveis, e propenso a muitos bugs, nos quais um programador desatento pode deixar uma brecha para uma invasão.
- Segundo ARSYS Internet S.L.(2006), Poucos são os inconvenientes que podemos

encontrar com esta linguagem de script. Tratam-se mais bem de pontos menos favoráveis, mas em nenhum caso, em desvantagem face ao uso de scripts em Perl.

Um destes pontos é:

- A depuração de erros, ainda que comum em todas as linguagens de script (e mais destacado no próprio Perl).

O outro é:

- O fato de o PHP ser uma linguagem especificamente concebida para a criação de scripts web, faz que esteja em desvantagem para realizar outras tarefas, em relação às linguagens de propósito gerais como Perl. No entanto, no desenvolvimento de aplicações web, o PHP possui um ótimo desempenho.

Uma das grandes desvantagens do php é realmente a questão da mistura dos códigos de php com as tags do html, sendo separados apenas por blocos php. Em contrapartida disto, pode-se utilizar uma arquitetura chamada “MVC<sup>1</sup>”.

## 6.2.2 PHP4

Oficialmente lançada em Maio de 2000, quase dois anos após o seu predecessor, o PHP 3.0. Além do melhoramento da performance, o PHP 4.0 incluiu outras características chaves como o suporte para varios servidores WEB, sessões HTTP, buffer de saída, maneiras mais seguras de manipular input de usuários e muitas construções novas na linguagem. Implementou principalmente as características de Orientação Objeto.(The PHP Group.2005).

---

1 MVC está descrito em maiores detalhes no capítulo que trata do comparativo entre frameworks.

### 6.2.3 PHP5

Uma das principais características do PHP5 certamente é a implementação do conceito de programação orientada ao objeto em seu projeto. O que com certeza fez com que muitos programadores vissem o PHP com outros olhos. Até a versão 4 o PHP não suportava todas as características que uma linguagem de programação orientada objeto deve conter. Com a reescrita do núcleo da linguagem ocorrida na versão 5 isso mudou.

#### 6.2.3.1 Orientação Objeto e conceitos

Neste capítulo são apresentados alguns conceitos básicos de orientação objeto e sua relação com PHP.

“Objeto, representa uma coisa física, tangível, uma idéia ou conceito. Possui um estado (o que ele sabe) e um comportamento (o que ele é capaz de fazer, como ele reage a estímulos externos)”.(LOZANO,Fernando.2002).

Também pode-se afirmar que um objeto é qualquer coisa, real ou abstrata, a respeito do qual armazenamos dados e os métodos de manipulação destes dados.(MARTIN, James.1995).

Exemplo para instanciar objetos em PHP:

```
<?php
    // criar instancia
    $objeto = new NomeDaClasse
    // chamada de método
    $objeto->NomeDoMetodo();
?>
```

Tabela 2: (Dall'Oglio, Pablo.2005).

Classe: Uma classe é uma implementação de um tipo de objeto. Ela especifica uma estrutura de dados ou métodos operacionais permissíveis que se aplicam a cada um de seus objetos. (MARTIN, James. 1995).

“Classe é um "molde" para a criação de objetos, fornecendo o seu comportamento padrão e a definição de todos os seus estados possíveis”. (LOZANO, Fernando. 2002).

Exemplo:

```
<?
# Classe Pessoa
Class Pessoa
{
    function Pessoa($nome)
    {
        $this->nome = $nome;
    }
    function GetNome()
    {
        return $this->nome;
    }
}

# Função CriaPessoa
function CriaPessoa($nome)
{
    return new Pessoa($nome);
}
?>
```

Tabela 3: (Dall'Oglio, Pablo. 2005).

Herança: Ocorre quando uma classe filho herda atributos de uma classe pai.

Herança ou Especialização. Uma nova classe pode ser definida em termos de uma

classe pai, herdando o seu comportamento. A nova classe especializa a classe pai, definindo apenas onde o seu comportamento deve ser diferente. (LOZANO,Fernando.2002).

Exemplo:

```
<?php
    include "ContaCorrente.php";

    $conta = new ContaCorrente (1000.0);
    echo ("Saldo inicial: {$conta->saldo}<BR>");
    $conta->saque (150.0);
    echo ("Novo saldo: {$conta->saldo}<BR>");
?>

Include_once "ContaCorrente.php";

class ContaEspecial extends ContaCorrente
{
    var $limite;
    function ContaEspecial ($valor, $limite) {
        $this-> ContaCorrente ($valor);
        $super ($valor);
    }
    function saque ($valor) {
        if ($this->saldo + $this->limite ->= $valor)
            $saldo = $valor;
    }
}
```

Tabela 4: (Dall'Oglio, Pablo.2005).

Encapsulamento: Encapsulamento é o ato de ocultar do usuário os detalhes de implementação de um objeto. (MARTIN, James.1995).

Exemplo:

```

<?php
class Predio {
    private $cor;
    private $coresPossiveis = array("red", "green", "blue");

    function mudaCor($cor) {
        if (in_array($cor, $this->coresPossiveis)) {
            $this->cor = $cor; # muda o valor da propriedade
            # deste objecto.
        }
    }
}
$umPredio = new Predio();
$umPredio->mudaCor("yellow"); //não altera o objecto
$umPredio->mudaCor("red"); //altera o objecto
?>

```

Tabela 5: (Morais, João Cruz.2004).

“Instância é uma ocorrência particular, identificada, de um objeto de uma determinada classe, com seu estado particular, independente de outras instâncias da mesma classe”.(LOZANO,Fernando.2002).

Exemplo:

```

<?php

$instance = new UmaClasseQualquer();
$atribuido = $instance;
$referencia =& $instance;

$instance->var = '$atribuido vai ter este valor.';
$instance = null; // $instance e $referencia ficam null

var_dump($instance);
var_dump($referencia);
var_dump($atribuido);
?>

NULL

NULL
object(UmaClasseQualquer)#1 (1) { ["var"]=> string(30) "$atribuido vai ter este
valor." }

```

Tabela 6: (Morais, João Cruz.2004).

“Polimorfismo. A mesma mensagem, quando enviada para objetos de classes diferentes, executa código particular da classe, mesmo que quem enviou a mensagem não tenha conhecimento do tipo específico de objeto sendo referenciado”.

(LOZANO,Fernando.2002).

Exemplo:

```
<?php
    class Relogio {
        function descricao() {
            echo "Indico as horas que são... agora!";
        }
    }
    class Livro {
        function descricao() {
            echo "Uma série de páginas com frases escritas";
        }
    }
    $objectos = array(new Relogio(), new Livro());
    foreach($objectos as $objecto)
        echo $objecto->descricao();
?>
```

Tabela 7: (Morais, João Cruz.2004).

“Agregação e Composição. Objetos podem conter outros objetos como partes constituintes, imitando o mundo real onde objetos são construídos em função de outros objetos. Podemos ou não expor as partes constituintes como parte da interface de um objeto”.

(LOZANO,Fernando.2002).

### 6.3 QUALIDADE DE SOFTWARE.

Uma boa definição de qualidade de software à alguns anos atrás seria simplesmente dizer que foram efetuados testes exaustivos em todos os componentes do software. Sendo que esta seria a última fase do desenvolvimento do mesmo.

Porém com a evolução da cultura da qualidade em produtos o software também teve seus aprimoramentos. Neste sentido foram criadas várias metodologias de controle de qualidade em softwares o que facilitou o trabalho dos desenvolvedores.

Segundo BARTIÉ, Alexandre,(2002), hoje em uma definição bem simples podemos dizer que a qualidade do software pode ser descrita desde a fase do projeto até a venda ao cliente, pois qualquer “alteração” ali existente, pode alterar todo o ciclo do projeto ocasionando desde um alargamento no prazo de entrega até um cancelamento do desenvolvimento por motivos de inviabilidade econômica do projeto. Para que isso não ocorra mais com tanta frequência como era de costume das empresas de software, buscou-se uma adequação as tendências de qualidade que o mercado exigia.

Ainda Segundo BARTIÉ, Alexandre,(2002), para que se torne possível obter um software de qualidade não podemos ter processos de desenvolvimento frágeis ou deficientes, então para que possamos estabelecer um processo de garantia da qualidade do software devemos focar o produto tecnológico e o processo de desenvolvimento. Podemos assim fixar duas dimensões principais para atingir a qualidade do software: Qualidade do processo e Qualidade do produto.

### **6.3.1 Qualidade do processo.**

Para poder garantir a qualidade do software, precisamos estruturar processos que possuam mecanismos de inibição e impedimento de falhas, possibilitando a identificação

prematura de defeitos. Nisto devemos incluir todas as partes do processo de desenvolvimento, desde os requisitos levantados até as projeções financeiras. Ou seja, todos e quaisquer documentos gerados durante o processo de desenvolvimento.

### **6.3.2 Qualidade do produto.**

A dimensão da qualidade do produto é baseada principalmente sobre os testes aplicados aos produtos. Cada empresa trabalha segundo sua própria metodologia empregando os tipos de testes que mais lhe convêm. A tática mais comum é usar uma fase específica no projeto para a execução dos testes, o que leva aos envolvidos uma série de processos que visam estressar telas e outras funcionalidades do produto.

Porém o grau de eficiência dos testes aplicados é muito baixo, o que leva a algumas empresas a substituírem isso por correção de problemas. Para que isso não ocorra se faz necessário:

- Fazer um planejamento maior sobre os testes.
- Testar as funcionalidades antigas e atuais.
- Criar mecanismos de automação.
- Fazer conferência com a equipe sobre os testes.

#### **6.3.2.1 Testes de verificação**

Fazem-se necessários a fim de avaliar toda a documentação gerada a cada etapa nova do processo. Se os documentos apresentarem um nível de defeito que não satisfaça a qualidade esperada a fase pode ser revista.

### 6.3.2.2 Testes de validação

Estes testes são feitos em vários estágios das aplicações com a finalidade de validar a estrutura da unidade dos softwares e sua aderência aos requisitos pré-estabelecidos. Ocorre também uma avaliação da integração com as demais unidades do software, validando a suas interfaces e seus componentes.

No final se valida à aplicação como um todo. Submetendo todos os testes possíveis.

Quanto maior a automação no processo de testes de softwares maior é o nível de cenários simulados, por consequência menor o nível de defeitos que podem furar o testes e maior é o grau de qualidade que pode ser obtida .

### 6.3.3 PHPUnit - unit testing framework for PHP

O PHPUnit é um derivado do Xunit<sup>1</sup>, que é um framework para o desenvolvimento, execução e tabulação dos testes.

O Xunit foi portado para diversas linguagens de programação, como o PHP. Um conjunto de testes é uma subclasse de TestCase, que executa código das classes do programa e verifica se os resultados são os esperados.

Instâncias de TestCase são adicionadas a TestSuite, que executa os testes e tabula os resultados.

TestSuite podem conter outros TestSuite de modo a facilitar a organização dos testes em hierarquias.(LOZANO,Fernando.2002).

O Xunit

- Todos os testes de unidade devem completar 100% para que uma integração seja bem sucedida.

- Todos os testes de funcionalidade devem completar 100% para que uma interação seja considerada completa.

O Xunit é um framework para o desenvolvimento, execução e tabulação dos testes.(LOZANO,Fernando.2002).

Exemplo de teste case:

---

1 O Xunit pode ser visto em maiores detalhes em: <http://www.lozano.eti.br/palestras/oo-php.pdf>

```

class TestesContaCorrente extends TestCase
{
    function TestesContaCorrente ($nome) {
        $this>TestCase ($nome);
    }
    function testaDeposito () {
        $conta = new ContaCorrente (1000.0);
        $conta>deposito (100.0);
        $this>assertEquals ($conta>saldo, 1100.0);
    }
    function testaSaque () {
        $conta = new ContaCorrente (1000.0);
        $conta>saque (2000.0);
        $this>assertEquals ($conta>saldo, 1000.0);
    }
}

```

Tabela 8: (LOZANO,Fernando.2002).

Exemplo TesteSuite:

```

<?php
    require "phpunit.php";

    // ... definição dos TestCases, provavelmente include

    $suite = new TestSuite;
    $suite>addTest (new TestSuite ("TestesContaCorrente"));

    // não é preciso fazer mais nada para que os testes sejam
/    / executados e tabulados em uma página HTML
?>

```

Tabela 9: (LOZANO,Fernando.2002).

## 6.4 FRAMEWORKS PHP

Segundo (CARNEIRO,Rafael). Se o framework estiver pronto, os benefícios são claros em termos de:

- Redução de custos
- Redução de time-to-market

Motivos:

- Maximização de re-uso (análise, design, código, testes)
- Desenvolvedores se concentram em adicionar valor em vez de reinventar a roda
- Menos manutenção
  - Fatoração de aspectos comuns a várias aplicações
  - Uso de herança permite corrigir todas as aplicações com a troca de uma classe-mãe.
  - Estabilização melhor do código (menos defeitos) devido ao uso em várias aplicações
  - Melhor consistência e compatibilidade entre aplicações

Ainda segundo (CARNEIRO,Rafael), os frameworks possuem algumas desvantagens tais como:

- Construir um framework é complexo o Re-uso não vem sozinho: deve ser planejado.
- É mais complexo e demora mais fazer uma aplicação tendo que construir um framework em vez de fazer a aplicação do zero.
- Benefícios são realizados em longo prazo, quem pode pensar em longo prazo quando se está competindo "On Internet time"? Poucas empresas.

- Precisa modificar o processo de desenvolvimento e criar novos incentivos.

Neste capítulo são descritos os principais frameworks para desenvolvimento usando PHP.

#### **6.4.1 PRADO - PHP 5 Framework**

“O PRADO é um PHP 5 Framework baseado em componentes e eventos, que torna o modelo de programação WEB muito similar ao ASP.NET. Criado por Qiang Xue, o PRADO foi inicialmente inspirado no projeto Apache Tapestry<sup>1</sup>. Durante o design e implementação, o Borland Delphi e o ASP.NET tiveram um papel importante na definição do Framework. Aqueles que já conhecem essas tecnologias vão se sentir mais confortáveis no seu entendimento”.(FRAMEWORK Prado.2005).

“O PRADO ainda possui vários recursos como (viewstate, sessions, caching, validação de formulários, autenticação e autorização)<sup>2</sup>. Além de tornar possível a criação de componentes de forma simples e prática”.(Framework Prado.2005).

Assim como o ASP.NET O Prado Framework utiliza uma técnica que define um campo escondido chamado “VIEWSTATE” e coloca no seu valor toda a informação de estado da página, com isso obtém-se algumas vantagens abaixo citadas:

- Não utiliza recursos do servidor, já que os valores ficam na página no cliente
- É compatível com qualquer navegador, já que os campos escondidos são um recurso antigo do HTML.
- Os valores a serem armazenados no campo VIEWSTATE são codificados em uma

---

<sup>1</sup> **Tapestry** é um framework open-source para criar dinamicamente aplicações altamente escalavel para web em Java.

<sup>2</sup> Maiores detalhes no decorrer do capítulo e no capítulo que trata do comparativo.

string juntamente com um “checksum<sup>1</sup>” para detectar eventuais alterações no conteúdo. (SANT’ ANNA, Mauro. 2002).

Ainda segundo SANT’ ANNA, Mauro. (2002), O uso de VIEWSTATE é comandado por propriedades *EnableViewState*, presente tanto no formulário como nos componentes individuais. Caso a propriedade *EnableViewState* do formulário seja desligada, nenhum componente manterá estado.

O Prado Framework ainda oferece o suporte a sessões, através da chamada do ID pelo usuário, a cada acesso é guardado um cookie<sup>2</sup> no lado do usuário contendo o seu ID. O que permite construir aplicações mais personalizadas e atrativas. ( THE PHP GROUP. 2006 ).

PRADO fornece uma técnica caching genérica usadas em diversas partes do núcleo da sua estrutura. Por exemplo, quando o caching é permitido, a classe `TTemplateManager` conservará os moldes analisados gramaticalmente. (Pradosoft Component Framework for PHP 5. 2006).

Segundo FRAMEWORK PRADO. (2005), O PRADO fornece os seguintes benefícios para os desenvolvedores:

- Reusabilidade - os códigos dos componentes de PRADO são altamente reusáveis. Tudo em PRADO é um componente reusável.
- Facilidade de utilização - criar e usar componentes são extremamente fáceis. Geralmente envolvem simplesmente configurar propriedades componentes.
- Robusto – PRADO reduz o esforço empreendido pelos colaboradores na criação de mais código, codificam nos termos dos objetos, métodos e propriedades, em vez de URLs e de parâmetros da pergunta.

---

1 **CHECKSUM:** um número de bits transmitido com os dados para que o dispositivo receptor possa verificar a precisão dos dados recebidos. Se o número de bits que chega é o mesmo enviado, a transmissão é considerada concluída.

2 **Cookies** são um mecanismo para guardar dados no navegador remoto possibilitando o acompanhamento ou identificação de usuários que retornam.

- Fornece um mecanismo de relatório de erro mais preciso.
- Desempenho - PRADO usa uma técnica de caching para assegurar o desempenho das aplicações baseadas nele.
- Integração da equipe - PRADO permite a separação do índice e da apresentação. Os componentes, tipicamente páginas, têm seu índice (lógica) e apresentação armazenada em locais diferentes.

Ainda segundo FRAMEWORK PRADO.(2005),50 a 75% do trabalho de uma aplicação web é realizado para gerar front end e validar os dados fornecidos pelos usuários.

Entre seus principais recursos estão :

- Html separado do código php
- Alto nível de reusabilidade por utilizar o conceito de componentes.
- Componentes para validação de formulários
- Suporte a módulos
- Arquivos em XML define a configuração da aplicação dos módulos e dos componentes.
- Suporte a internacionalização.
- Recursos de cache para aumentar a performance da aplicação.

#### **6.4.2 Symfony Framework**

“Symfony é uma estrutura PHP5 Orientado a objeto baseada no modelo MVC<sup>1</sup>.

Symfony permite a separação de regras de negócio, a lógica do usuário e a visão da

---

<sup>1</sup> **MVC** Esta arquitetura estabelece uma separação da estrutura em três partes distintas: Modelo, Vista e Controle. Em maiores detalhes no capítulo que trata do comparativo.

apresentação de uma aplicação para web. Também contem ferramentas numerosas e classifica-as visado encurtar a fase de desenvolvimento de uma aplicação complexa para web." (H3RALD.2006).

Fatores favoráveis:

- A estrutura inteiramente caracterizada, inclui tudo que o programador da web necessita.
- O suporte completo e nativo a internacionalização.
- Boa documentação, com: tutoriais, wiki's<sup>1</sup>, livros, os screencasts<sup>2</sup>, APIs, e exemplos reais.
- Geradores de códigos.
- Vários módulos e bibliotecas "pré-construídas" para as tarefas mais comuns.
- Foi inspirado pelos melhores conceitos e práticas de vários outros frameworks.
- Bom suporte da comunidade.

Fatores desfavoráveis a sua utilização:

- Parece ser muito grande comparado aos outros, e com muitos recursos que não são úteis a todos os colaboradores.(H3RALD.2006).
- Contempla somente o PHP5 .( H3RALD.2006).
- Não é recomendado para projetos simples. (H3RALD.2006).

---

1 **Wiki** é um sistema de site que permite a edição do seu conteúdo por qualquer um. Não só se adiciona conteúdo, como se complementa o conteúdo existente, editando-o, acrescentando ou corrigindo.

2 **Screencasts** : É um processo de criar demonstrações e simulações interativas de softwares através de uma série de telas de uma aplicação.

Symfony implementa a execução fácil de AJAX<sup>1</sup> e inclui o suite inteiro de scripts .”script aculo.us”<sup>2</sup> de efeitos do Javascript. Symfony tem também a habilidade de gerar “CRUD e scaffolding”<sup>3</sup> da aplicação de uma base de dados já construída em SQL. Isso significa que ele incorpora toda SQL.(PULIDO,Nick.2006).

Bases de dados suportadas: MySQL, PostgreSQL, SQLite, Oracle, MS SQL e qualquer outra suportada pela camada de abstração da base de dados Creole<sup>4</sup>.(H3RALD.2006).

### 6.4.3 Cake PHP Framework

O Cake é um framework afiado no desenvolvimento rápido de aplicações. E na fácil execução de AJAX.(PULIDO,Nick.2006).

Cake é um Framework rápido de desenvolvimento para PHP que usa padrões como ActiveRecord<sup>5</sup>.

Este modelo fornece uma estrutura que permite aos usuários de PHP em todos os níveis o poder de desenvolver rapidamente aplicações robustas para web, sem nenhuma perda à flexibilidade. (H3RALD.2006).

---

1 **AJAX** é um termo que descreve uma técnica de desenvolvimento web para criar softwares de interação. Em mais detalhes no cap 5.

2 **Script.aculo.us** fornece consigo bibliotecas de fácil utilização, e compatíveis com o Javascript. São bibliotecas para desenvolver interfaces mais ricas.

3 Um **scaffold** é um meio de criar código para um determinado modelo (que indicamos) através de um determinado controlador (que indicamos também). É um meio de começarmos rapidamente a ver os resultados no nosso navegador web e um método muito rápido de implementar o CRUD (Create, Retrieve, Update, Delete) na sua aplicação. Lembrando que o scaffold cria código que, fatalmente, vai ser alterado depois, a não ser que você deseje manter o nível muito básico de interface e controle padrão de campos que ele proporciona.(Rangel, Eustáquio).

4 **Creole** é uma camada de abstração de bases de dados para php5.

5 **Active Record** efetua automaticamente a correspondência entre tabelas e classes, linhas a objetos (exemplos das classes modelos), e colunas para atributos de objetos.

Fatores favoráveis:

- Contém somente o código essencial.
- Funciona em PHP4 e PHP5.
- É necessária somente a execução de uma configuração simples e curta da base de dados e em algumas constantes que podem ser modificadas. Você pode literalmente começar programar em menos de cinco minutos.
- Permite a criação de arquiteturas complexas da base de dados.
- Estrutura extremamente lógica e funcional de diretórios.
- Sustentação realçada de AJAX através dos ajudantes que auxiliam na criação de AJAX e de Javascript.
- Possui um script de linha de comando para gerar automaticamente partes do código, chamado BAKE.
- Comunidades muito ativas.
- Adequado para qualquer tipo de website, da aplicação pessoal de pequeno porte à aplicação avançada de e-business.

Segundo H3RALD.(2006), podemos citar alguns fatores desfavoráveis em relação ao

Cake:

- Nenhuma sustentação "oficial" de internacionalização para a versão atual, mas será incluída no passo seguinte.
- Não faz uso inteiramente das vantagens oferecidas pelo PHP5.
- A documentação oficial necessita ainda alguma melhoria, embora agora pareça

consideravelmente completa e exaustiva.

Bases de dados suportadas: MySQL, PostgreSQL, SQLite, MS SQL + e outras que suportem a camada de abstração das bases ADOdb ou PEAR::DB.

#### **6.4.4 Requisitos de uso dos frameworks**

Para fazer uso de um framework de PHP você precisa de algumas ferramentas tais como um servidor HTTP rodando e o PHP instalado. Com exceção do Cake Framework que roda também com o php4 as outros dois frameworks analisados necessitam que esteja rodando o PHP5. Tendo então estes aplicativos corretamente instalados e configurados, basta ir até o site do seu framework preferido e baixar o pacote do framework para sua máquina, e então descompacta los no diretório root do seu servidor web, no meu caso /var/www/html/ e fazer o seu acesso via browser no endereço [http://localhost/Nome\\_do\\_framework/](http://localhost/Nome_do_framework/).

Em geral todos os frameworks oferecem suporte as mais variadas bases de dados existentes. Resta então ao usuário definir-se por alguma delas e proceder a instalação, sempre lembrando o mesmo de verificar a compatibilidade do framework com a base de dados escolhida, para evitar posteriores problemas.

#### **6.5 TÓPICOS PARA COMPARAÇÃO**

### 6.5.1 Tabela comparativa

Framework	PHP4	PHP5	MVC	Multiple DB's	ORM		
CakePHP	Sim	Sim	Sim	Sim	Sim		
Symfony Project	Não	Sim	Sim	Sim	Sim		
Prado	Não	Sim	Parcialmente	Sim	-		
Framework	Validation	Ajax	Auth Module	Modules	Tableless		
CakePHP	Sim	Sim	Sim	-	-		
Symfony Project	Sim	Sim	Sim	-	Sim		
Prado	Sim	Sim	Sim	Sim	-		
Framework	DB Objects	Templates	Caching	Internacionalização	Integração Pear	Documentação detalhada	
CakePHP	Sim	-	Sim	Não	-	Parcialmente	
Symfony Project	Sim	Sim	Sim	Sim	Sim	Sim	
Prado	-	Sim	Sim	Sim	-	Parcialmente	

Tabela 10 : (Comunidade Cyaneus.2006).(The Pallett Group.2005).

Os principais tópicos utilizados para comparação entre os frameworks estão ilustrados no quadro abaixo:

A seguir são descritos os principais tópicos analisados.

#### 6.5.1.1 Mvc

A arquitetura MVC foi desenvolvida para ser usada no projeto de interface visual em Smalltalk<sup>1</sup>. Esta arquitetura estabelece uma separação da estrutura da aplicação em três partes distintas: Modelo, Vista e Controle.

- Vista gerencia a saída gráfica e textual da parte da aplicação visível ao usuário.
- Controle interpreta as entradas de mouse e teclado do usuário, comandando a Vista e o Modelo para se alterarem de forma apropriada.

---

1 Smalltalk é uma linguagem de programação Orientada a Objetos criada na década de 70 pelo Centro de Pesquisa de Palo Alto da Xerox.

- Modelo gerencia o comportamento e os dados do domínio da aplicação, responde as requisições sobre o seu estado (geralmente vindas da Vista), e responde às instruções para mudança de estado (geralmente vindas do Controle).  
(BURBECH,2002).

#### **6.5.1.2 Multiples DB's**

Todos os frameworks pesquisados oferecem suporte a múltiplas DB's como descrito nos mesmos.

#### **6.5.1.3 ORM**

Object Relational Mapping ou "Mapeamento Objeto Relacional" é uma técnica de desenvolvimento utilizada para reduzir a impedância do desenvolvimento orientado ao objeto utilizando bancos de dados relacionais. As tabelas do banco de dados são representadas como objetos. Com esta técnica, é possível associar tabelas dos bancos de dados relacionais com classes de uma linguagem de programação orientada a objeto, utilizando tecnologia XML\_ de maneira que o programador não precise se preocupar com os comandos em linguagem SQL, mas simplesmente utilize uma Interface de Programação simples (API) que faça todo o trabalho de persistência. (OBJECT-RELATIONAL MAPPING.2006).

#### **6.5.1.4 Validation**

Indica se o framework tem alguma estrutura de validação embutida em sua estrutura.

### **6.5.1.5 AJAX**

AAX(Asynchronous JavaScript and XML) é um termo que descreve uma técnica de desenvolvimento web para criar softwares de interação. A tecnologia usa combinações do HTML, XML e CSS para apresentar as informações, trocando dados assincronamente com o servidor web.

Entre algumas vantagens de seu uso estão:

- Mais acessível.
- Maior controle sobre o consumo da memória da máquina do usuário.
- Não precisa de plugin.

A grande diferença entre o AJAX e o modelo e as aplicações até então usadas é que o modelo convencional o usuário envia uma requisição ao servidor que por sua vez recebe e processa o código, conversa com outros sistemas paralelos e retorna ao servidor de internet que publica o código em forma de HTML. Com o AJAX isso acontece de forma assíncrona ou seja enquanto o usuário está visualizando ou utilizando algo o site, são feitas inúmeras requisições ao servidor sem prévio aviso. Fazendo com que a exibição do site se torne uma visualização em tempo real. (VIEIRA,Victor.2006 ).

### **6.5.1.6 Auth Module**

Indica se a estrutura tem um módulo embutido para assegurar a autenticação do usuário.

### **6.5.1.7 Modules**

Indica se a estrutura do framework é modularizada por exemplo, possui um parser da alimentação de RSS<sup>1</sup>, o módulo de separação de pdf's ou etc;

#### **6.5.1.8 Tableless**

O nome Tableless é um nome mais “publicitário” para se referir a sites que seguem os Padrões. Os sites Tableless não são construídos usando as famigeradas tables. Elas usam XHTML para apresentar a informação e as Folhas de Estilo (CSS) para formatar essa informação. Pelo motivo de as tables não serem usadas para a estruturação, essa metodologia se chama Tableless.(TABLELESS.2006).

#### **6.5.1.9 DB Objects**

Indica se a estrutura inclui outros objetos da base de dados, como classes de linguagem de programação orientada a objetos.

#### **6.5.1.10 Templates**

Segundo MINETTO,Elton.(2006),Qualquer webmaster que enfrentou a missão de manter um site de médio a grande porte, sabe a dificuldade que isso significa, principalmente no que diz respeito a relação entre o design e a programação. Alterar dezenas de páginas cada vez que um detalhe de design, como uma cor de fundo ou uma imagem, é alterado é uma tarefa desgastante. Mas, para alívio de todos, existe uma solução para o problema e esta solução está nos templates.

Os templates, ou modelos, têm por finalidade separar o design, a parte HTML, da programação, no nosso caso, o PHP.

#### **6.5.1.11 Caching**

---

1 RSS é um sub-conjunto de "dialetos" XML que servem para agregar conteúdo . É usado principalmente em sites de notícias e blogs.

Caching é o melhor recurso para a redução de processamento do servidor envolvendo a renderização e lógica da camada de acesso a dados.

Podemos dividir o caching em três tipos básicos:

- Cache de Respostas (OutputCache)

O cache de respostas é a forma mais simples de se tirar proveito do sistema de cache, sem necessidade de redesenho ou alterações de código - o conteúdo a ser enviado ao cliente (resposta) é armazenado em memória e disponibilizado para as próximas requisições. De fato, todo o conteúdo dinâmico pode ser armazenado em mecanismos que suportem o HTTP 1.1 (servidor web, navegadores e proxies) de forma que requisições subsequentes sejam servidas direto do cache sem execução de código.( FERREIRA, Miguel.2006).

- Cache de Fragmento

Existem situações onde o “cacheamento” de toda uma página de resposta não é adequado. Um caso óbvio, seria àquele no qual o desenho da aplicação requerer que algumas partes/seções de uma página específica tenham versões específicas para um determinado grupo de usuários (e.g. menu baseado no perfil do usuário) e que outras partes sejam comuns à todos (e.g. conteúdo genérico c/ notícias gerais). Este é o cenário ideal para o uso do fragment caching.(FERREIRA, Miguel.2006).

- Cache de Objetos

O objeto de Cache é recriado a cada reinício da aplicação, o que nos lembra uma certa similaridade com o objeto Application. A principal diferença entre estes os dois reside no fato de que o objeto de Cache proporciona funcionalidade específicas, tais como dependências e políticas de expiração.( FERREIRA, Miguel.2006).

#### **6.5.1.12 Internacionalização**

Suporte a internacionalização, ou seja, criar sites em vários idiomas apenas trocando os arquivos de propriedades, sem ter que mexer no fonte.

### 6.5.1.13 Integração Pear

Pear é um sistema de repositório e de distribuição para componentes reusáveis de php.

Pear é composto por:

- Repositório organizado e padronizado de código aberto para programadores de PHP

Sistema para a distribuição e manutenção de pacotes.

- PFC (*PHP Foundation Classes*).

- Estilo padronizado de código.

- Sítos e listas de discussão para a comunidade de colaboradores.(PEAR THE PHP GROUP.2005).

O código em PEAR é dividido em “pacotes”. Cada pacote é um projeto separado com sua própria equipe do desenvolvimento, número de versão, ciclo da liberação, documentação e uma relação definida a outros pacotes (dependências including).(PEAR THE PHP GROUP.2005).

### 5.1.14 Documentação detalhada

Uma boa documentação não é somente uma descrição do que é a ferramenta e para que ela serve. Mas sim toda descrição histórica, todo um apoio para os desenvolvedores como exemplos práticos de sua aplicação, detalhes da sua criação e implementação, descrição do comportamento e funcionamento da mesma, fóruns de discussão, e demais conceitos abordados acima.

## 7 .CRONORAMA DE AÇÃO

Abaixo seguem cronogramas do Trabalho de Conclusão Curso (TCC) I (Quadro 1) e do Trabalho de Conclusão de Curso II (Quadro 2).

**Quadro 1 . Cronograma de Ação do TCC I (1º semestre 2006)**

Atividades /Meses	Fev	Mar	Abr	Mai	Jun
Elaboração das Etapas Iniciais do Projeto	X				
Pesquisa Bibliográfica Necessária	X	X	X	X	
Revisão Ortográfica e Formatação				X	X
Entrega para pré-banca					X
Apresentação					X

Abaixo segue o cronograma para TCC II

**Quadro 2 . Cronograma de Ação do TCC II (2º semestre 2006)**

Atividades	Ago	Set	Out	Nov	Dez
Revisão Bibliográfica	X	X	X	X	
Desenvolvimento do Protótipo		X	X	X	
Confecção do Artigo			X		
Submissão do Artigo			X		
Testes		X	X		
Validações			X		
Entrega				X	
Apresentação				X	
Realização das correções sugeridas pela banca					X

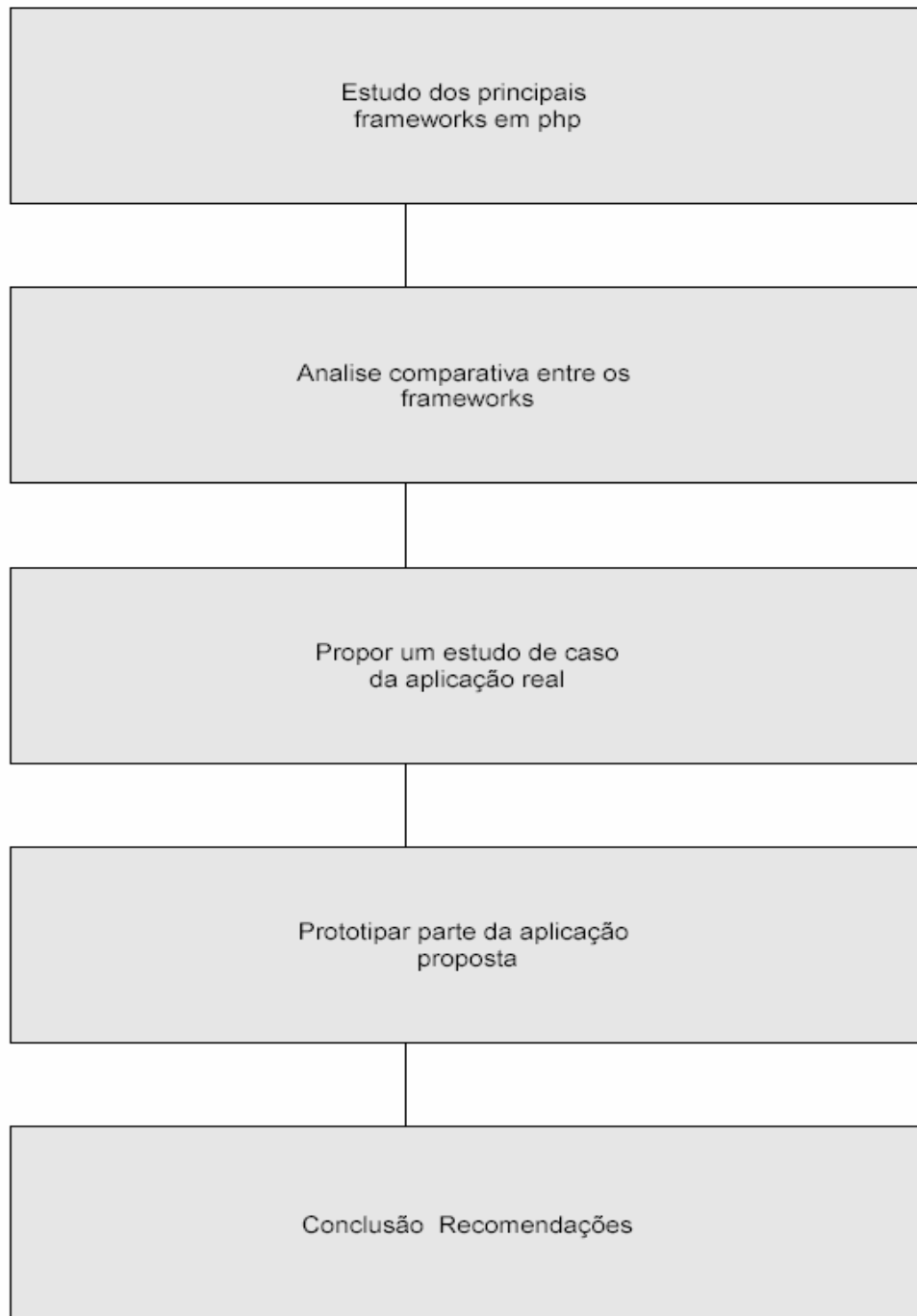
## **8 .ORÇAMENTO**

- Todo o trabalho foi desenvolvido utilizando recursos próprios do acadêmico e em computadores disponibilizados pela Unochapecó nos laboratórios de informática.
- Despesas com papéis, impressões (Custeadas pelo acadêmico).
- Softwares que podem ser obtidos através da Internet:
  - Frameworks que são livres.
  - Linguagem de programação PHP;
  - Servidor Apache;
  - Banco de dados Relacional;
- Muitas horas de sono. (Custeadas pelo acadêmico).
- Remédio para Gastrite e similares. (Custeadas pelo acadêmico).

## **9 .PROCEDIMENTOS METODOLÓGICOS**

A pesquisa a realizada no presente trabalho é de natureza básica, pois serão analisadas e comparadas duas ou mais tecnologias e suas formas de implementação através de um protótipo, onde serão abordadas de uma forma qualitativa de informações com fins exploratórios de pesquisa bibliográfica.

A figura abaixo auxilia o entendimento do contexto do trabalho.



*Tabela 11( Procedimentos Metodológicos).*

## 10 .REFERÊNCIAS BIBLIOGRÁFICAS

DMITRUK, Hilda Beatriz (Org). Cadernos Metodológicos: Diretrizes do trabalho científico . 6. ed. rev. ampl. atual. Chapecó: Argos, 2004.

BARTIÉ, Alexandre. Garantia da qualidade de software: adquirindo maturidade organizacional. Rio de Janeiro: Editora Campus, 2002.

Martin,James. Análise e projeto orientados a objeto/ James Martin, James J. Odell; tradução José Carlos Barbosa dos Santos; revisão técnica Ronald Stevis Cassiolato. -São Paulo: Makron Books, 1995.

De Souza,Marcos Vinícius Bittencourt. Estudo Comparativo entre Frameworks Java para Construção de Aplicações Web.2004. Acessado em 27/04/2006 disponível em: <http://www-usr.inf.ufsm.br/~marvin/monografia.pdf>

Cavalcanti,Éric, PRADO - PHP 5 Framework.2004 Acessado em 01/05/2006 disponível em: [http://www.linhadecodigo.com.br/artigos/videos/eric\\_cavalcanti/prado/prado.html](http://www.linhadecodigo.com.br/artigos/videos/eric_cavalcanti/prado/prado.html)

Framework Prado.2005 Acessado em 02/05/2006 disponível em: <http://www.xisc.com/>

Wikipédia, a enciclopédia livre.2006 Acessado em 01/05/2006 disponível em: <http://pt.wikipedia.org/wiki/Framework>

Pradosoft Component Framework for PHP 5.2006 Acessado em 03/05/2006 disponível em: <http://www.pradosoft.com/>

Sant' Anna,Mauro,Mantendo estado em páginas ASP.NET.2002 acessado em 02/05/2006 disponível em: <http://72.14.203.104/search?q=cache:Gye7UImWOYMJ:www.mas.com.br/Artigos/mantendo.htm+viewstate&hl=pt-BR&gl=br&ct=clnk&cd=1>

The PHP Group.2006 Acessado em 03/05/2006 disponível em:  
[http://www.php.net/manual/pt\\_BR/introduction.php](http://www.php.net/manual/pt_BR/introduction.php)

h3rald.2006 Acessado em 05/05/2006 disponível em:  
<http://www.h3rald.com/articles/view/rails-inspired-php-frameworks>

Pulido,Nick The Web 2.0 Dev.2006 Acessado dia 05/05/2006 disponível em :  
<http://www.theweb20dev.com/wordpress/2006/05/03/5-next-generation-php-frameworks/>

BURBECH.2002. Acessado em 08/05/2006 disponível em: <http://hercules.nce.ufrj.br/arq-mvc.html>

Tableless.2006. Acessado em 08/05/2006 disponível em:  
<http://pt.wikipedia.org/wiki/Tableless>

Souza.S.J.Sandro.Vantagens do PHP 5 frente ao ASP.NET.2005 acessado dia 10/05/2006 disponível em : <http://phpbrasil.com/articles/article.php/id/1173>

Wikipédia, a enciclopédia livre.2006 acessado em 11/05/2006 disponível em:  
<http://pt.wikipedia.org/wiki/PHP#column-one>

Comunidade Cyaneus.2006. Acessado em 14/05/2006 disponível em:  
<http://4h17.cyaneus.net/frameworksphp>

The Pallett Group.2005. Acessado em 14/05/2006 disponível em:  
<http://www.phpit.net/article/ten-different-php-frameworks/1/>

Canal Html.2006. Acessado em 15/05/2006 disponível em:  
<http://www.htmlstaff.org/entrevistas/entrevistas24.php>

Arsys Internet S.L.2006. Acessado em 15/05/2006 disponível em:  
<http://www.arsys.pt/suporte/programacao/linux.htm>

Object-relational mapping.2006. Acessado em 15/05/2006 disponível em:  
[http://en.wikipedia.org/wiki/Object-relational\\_mapping](http://en.wikipedia.org/wiki/Object-relational_mapping)

Lozano,Fernando Programação Orientada a Objetos com PHP.2002. Acessado em 18/05/2006 disponível em: <http://www.lozano.eti.br/palestras/oo-php.pdf>

Wikipédia, a enciclopédia livre.2006 Acessado em 19/05/2006 disponível em: <http://pt.wikipedia.org/wiki/PHP-GTK>

PHP-GTK Brasil.2006 Acessado em 19/05/2006 disponível em: <http://www.php-gtk.com.br/index.php?class=Home::HomeHtml>

The PHP Group.2005 Acessado em 03/05/2006 disponível em: [http://www.php.net/manual/pt\\_BR/history.php](http://www.php.net/manual/pt_BR/history.php)

Rangel, Eustáquio.Tutorial de Rails. Acessado em 22/05/2006 Disponível em: <http://plentz.org/unsorted/ruby/tutorialrails.pdf>

(VIEIRA,Victor. [Www.com.br](http://www.com.br). Por dentro do Ajax. Editora Europa. **São Paulo**, n.63,p.50-52,2006 ).

Ferreira, Miguel. ASP.NET Caching - Escalabilidade e desempenho à mão Acessado em 22/05/2006 Disponível em: <http://www.microsoft.com/brasil/msdn/tecnologias/aspnet/aspnetcache.aspx?mfr=true>

Dall'Oglio, Pablo.2005. As novidades do PHP5 Acessado em 23/05/2006 Disponível em: <http://www.dalloglio.net/events/php5.pdf>

Pear The PHP Group.2005 Acessado em 25/05/2006 disponibilizado em: <http://pear.php.net/>

Carneiro,Rafael. Acessado em 26/05/2006 Disponível em: <http://www.univel.br/roberta/FRAMEWORKS/frameworks%5B1%5D.pdf>

Morais, João Cruz. PHP Webpages Dinâmicas Acessado em 02/05/2006 Disponibilizado em: [http://64.233.187.104/search?q=cache:Iv0hyoMPHV4J:groups.tagus.ist.utl.pt/neiist/web/eventos/ca2/aps/php-mod2-alameda-14-out-2004/php\\_-\\_webpages\\_dinamicas\\_-\\_modulo2.pps+exemplos+de+encapsulamento+em+php&hl=pt-BR&gl=br&ct=clnk&cd=1](http://64.233.187.104/search?q=cache:Iv0hyoMPHV4J:groups.tagus.ist.utl.pt/neiist/web/eventos/ca2/aps/php-mod2-alameda-14-out-2004/php_-_webpages_dinamicas_-_modulo2.pps+exemplos+de+encapsulamento+em+php&hl=pt-BR&gl=br&ct=clnk&cd=1)

Manual do PHP Acessado em: 02/05/2006 disponível em: <http://ceres.inf.ufsc.br/doc/php-br/function.sqlite-create-aggregate.html>

Minetto,Elton Luís.2006. O uso de templates em PHP Acessado em 05/06/06 Disponibilizado em: [http://200.135.240.21/~elm/docs/php\\_templates.htm](http://200.135.240.21/~elm/docs/php_templates.htm)